NAME (as it appears on your UF ID): _____
(Please **PRINT**)

UF Student ID#: _____

---------- CEN 4072/6070 Software Testing & Verification ----------

Exam 2 – Spring 2013

You have 90 minutes to work on this exam. It is a "closed-book/closed-notes" test. Pay attention to point values, since you may not have time to complete all 11 problems. **You should assume that all variables represent INTEGER values, unless otherwise indicated.**

PRINT your name above NOW and sign the pledge at the bottom of the last page, if appropriate, when you are finished.

PLEASE PRINT ANSWERS IN THE SPACE PROVIDED ONLY – PREFERABLY USING A BALLPOINT PEN TO INCREASE LEGIBILITY. Good luck!

1. (12 pts.) Consider a program S, pre-condition {x>0}, and post-condition {y=z+x}. Which of the following observations/facts would allow one to deduce that S is NOT **strongly** correct with respect to the given pre- and post-conditions? Circle either "would" or "would not" as appropriate, considering the observations/facts individually. To compensate for random guessing, your score in points will be 2 times the number of [correct minus incorrect] answers, or 0 – whichever is greater. Therefore, if you are not more than 50% sure of your answer, consider skipping the problem.

a. For at least one initial value of x satisfying the given pre-condition, S terminates with y=z.	would	would not
b. sp(S, x=17) is undefined	would	would not
c. wlp(S, y=z) = x>5	would	would not
d. wp(S, true) = (x>-5 ∨ y=5)	would	would not
e. sp(S, true) = (x>0 ∧ y=z)	would	would not
f. [{x>0} S {y=z+x}] is false	would	would not

2. (20 pts.) Circle either "true" or "false" for each of the following assertions. To compensate for random guessing, your score in points will be 2 times the number of [correct minus incorrect] answers, or 0 – whichever is greater. Therefore, if you are not more than 50% sure of your answer, consider skipping the problem.

a. $\{x>0\}\ S\ \{x=17\} \Rightarrow \{x=17\}\ S\ \{x>0\}$ true false

b. $\{x=17\}\ S\ \{x>0\} \Rightarrow \{x>0\}\ S\ \{x=17\}$ true false

c. {true} while x <> 5 do x := x-1 {x=5} true false

d. Z=XJ is a **Q-adequate loop invariant** for the assertion: true false

```
{Z=X ∧ J=1}
    while J<Y do
        Z := Z+X
        J := J+1
    end_while
{Z=XY}
```

e. Z=XJ is a **loop invariant** for the assertion given in part (d) above. true false

f. The following program terminates for all initial values of x,y. true false

```
while (y<0) do
   y := y+x
   if (x≤0) then
        x := x+1
   end_if
end_while
```

g. The **Method of Well Founded Sets**, as presented in class, can be used to prove that the program given in part (f) above will terminate. true false

h. $\{P\}$ repeat S until b $\{Q\} \Leftrightarrow$ true false

```
{P}
   S;
   while NOT b do
      S
   end_while
{Q}
```

i. Suppose k = wp(repeat S until b, Q). Then k is a Q-adequate loop invariant for {k} repeat S until b {Q}. true false

j. Suppose k = wp(repeat S until b, Q). Then sp(S,k) is a Q-adequate loop invariant for {k} repeat S until b {Q}. true false

3. a. (2 pts.) Complete the following ROI for proving the correctness of program S with respect to pre-condition P and post-condition Q using the strongest post-condition (sp) predicate transform:

$$\frac{\qquad\qquad}{\{P\}\ S\ \{Q\}}$$

b. (4 pts.) Give the strongest post-condition (sp) transform rule for the program statement *if b then S* with respect to pre-condition P.

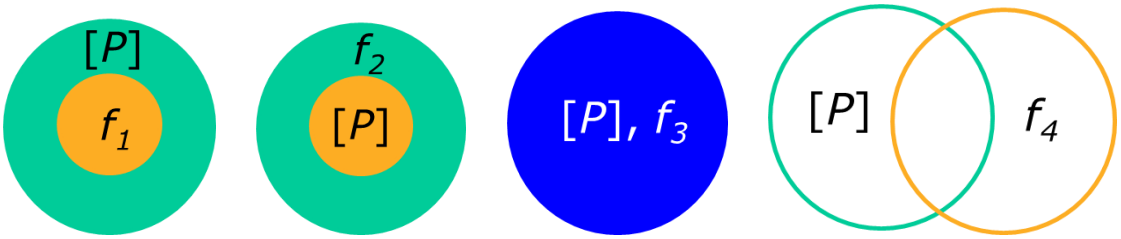
sp(if b then S, P) ≡ _____

c. (9 pts.) Use the ROI (a) and transform rule (b) to **prove** the following assertion:

$\{y=x\}$ if $y>0$ then $y := y-(2*y)$ $\{(y=-x \wedge y'=x) \vee (y=x)\}$

SHOW **ALL** TERMS, STEPS, AND NECESSARY CASES for the proof.

4. (8 pts.) The four diagrams below represent relationships between the function computed by a program, *P*, and 4 different intended functions, f_1, f_2, f_3, and f_4.



In the table below, indicate "C" for Complete program correctness, "S" for Sufficient program correctness only, and "N" for Neither. To compensate for random guessing, you will receive +2 pts. for each correct answer and -1 pt. for each incorrect answer, with a minimum possible score of 0 pts.

	f_1	f_2	f_3	f_4
P				

5. (9 pts.) Determine (but do not formally verify) the function of the following program:

```
y := x-2
if y>0 then
   while x<> 5 do
      x := x-1
   end_while
end_if
```

Give your answer in the simplified form: ($p_1 \rightarrow$ x,y := ?,? | $p_2 \rightarrow$ x,y := ?,?) where p_1 and p_2 are Boolean predicates, the union of which specify the function domain.

Hint: Use heuristics to identify the function of the while loop and then use the Axiom of Replacement and function composition to determine the program function. Also, write Boolean predicates and Identity functions explicitly (as in "y≤0 $\rightarrow$ x,y := x,y" instead of "true $\rightarrow$ I") to facilitate function composition.

program function: _____

6. (20 pts.) Circle either "valid" or "invalid" for each of the following *hypothesized* Rules of Inference. To compensate for random guessing, your score in points will be 2 times the number of [correct minus incorrect] answers, or 0 – whichever is greater. Therefore, if you are not more than 50% sure of your answer, consider skipping the problem.

a. $\dfrac{sp(S, true) \Rightarrow Q}{\{P\}\ S\ \{Q\}}$? valid invalid

b. $\dfrac{sp(S, P) \Rightarrow Q}{\{P\}\ S\ \{Q\}\ \text{strongly}}$? valid invalid

c. $\dfrac{P \Rightarrow (wlp(S, Q) \wedge K)}{\{P\}\ S\ \{Q\}}$? valid invalid

d. $\dfrac{\{P\}\ \text{if b then S}\ \{Q\}}{\{P\}\ \text{if b then S}\ \{Q\}\ \text{strongly}}$? valid invalid

e. $\dfrac{P \Rightarrow I,\ \{I \wedge \neg b\}\ S\ \{I\},\ (I \wedge b) \Rightarrow Q}{\{P\}\ \text{repeat S until b}\ \{Q\}}$? valid invalid

f. $\dfrac{(\neg b) \Rightarrow Q}{\{P\}\ \text{while b do S}\ \{Q\}\ \text{strongly}}$? valid invalid

g. $\dfrac{\{true\}\ \text{if b then S}\ \{\neg b \wedge Q\}\ \text{strongly}}{\{P\}\ \text{while b do S}\ \{Q\}}$? valid invalid

h. $\dfrac{\{P \wedge b\}\ S\ \{P\},\ (P \wedge \neg b) \Rightarrow Q}{\{P\}\ \text{while b do S}\ \{Q\}}$? valid invalid

i. $\dfrac{\{P\}\ S\ \{K\},\ \{K\}\ \text{while}\ \neg b\ \text{do S}\ \{Q\}}{\{P\}\ \text{repeat S until b}\ \{Q\}}$? valid invalid

j. $\dfrac{\{P\}\ S\ \{I\},\ (I \wedge b) \Rightarrow Q}{\{P\}\ \text{repeat S until b}\ \{Q\}}$? valid invalid

7. a. (6 pts.) Complete the following statement of the Invariant Status Theorem (IST), as presented in class, by filling-in the six missing predicates below:

Theorem.

Let f = [while p do g]. If $X_0 \in D(f)$, $X \in D(f)$, and $q(X)$ = (_____), then q is an invariant of while p do g; i.e., it has the following properties:

1. _____ is true, and
2. ($q(X) \wedge$ _____) $\Rightarrow$ _____.

In addition, $q(X)$ is an f-adequate invariant; i.e.,

3. ($q(X) \wedge$ _____) $\Rightarrow$ (_____).

b. (3 pts.) Consider function $\boldsymbol{t = (k \geq 0 \rightarrow p,k := p2^k,0).}$ Apply the IST to derive a t-adequate invariant, q, of t for $X_0 \in D(t)$, $X \in D(t)$. (You need not show your work – just give q.)

q: _____

c. (6 pts.) Consider program ***T*: while k<>0 do p := p∗2; k := k-1 end_while** designed to implement *t.* Assume that term*(t,T)* holds. Complete the proof that $t = [T]$ by showing that q from part (b) satisfies IST properties (2) and (3) from part (a). (Recall that property (1) is a tautology for $q(X)$ and therefore independent of program T.) Assume that the function of the loop body, g, is (p,k := 2p,k-1).

---------- please do not write below this line ----------

7. (cont'd)

d. (12 pts.) Prove $t = [T]$ by showing that the while_do complete correctness conditions hold. Again, you may assume term*(t,T)* and that the function of the loop body, g, is (p,k := 2p,k-1). SHOW AND JUSTIFY ALL STEPS AND CASES AS ILLUSTRATED IN CLASS.

$t = (k \geq 0 \rightarrow p,k := p2^k,0)$ **T: while k<>0 do p := p∗2; k := k-1 end_while**

7. (cont'd)

e. (10 pts.) Find the weakest pre-condition (wp) of program *T* with respect to the post condition **p=16**. Give the values of H_0, H_1, H_2, and $H_{n>0}$ in the space provided below, where the wp is given by the infinite expression: $H_0 \vee H_1 \vee H_2 \vee \ldots$ (Hint: don't forget that p and k are assumed to be INTEGER variables.)

***T*: while k<>0 do p := p∗2; k := k-1 end_while**

H_0:

H_1:

H_2:

$H_{n>0}$:

f. (4 pts.) Collapse the infinite expression obtained for part (e) into an equivalent FINITE (closed form) expression in terms of the program variables ONLY (i.e., with no reference to "n"). (Hint: Since p and k are assumed to be INTEGER variables, any predicate in the infinite expression $H_0 \vee H_1 \vee H_2 \vee \ldots$ implying that p or k has a non-integer value is necessarily FALSE and can therefore be dropped.)

g. (3 pts.) What is the weakest ***liberal*** pre-condition (wlp) of program *T,* in closed form, with respect to the post condition p=16?

8. (8 pts.) Consider the assertion:

$\{ p=1 \wedge k=n \}$

```
while k<>0 do
  p := p*2
  k := k-1
end_while
```

$\{ p=2^{n} \}$

Use the invariant I: $p=2^{n-k}$ and the While Loop Rule of Inference to prove the assertion. SHOW AND JUSTIFY ALL STEPS AND CASES AS ILLUSTRATED IN CLASS.

---------- please do not write below this line ----------

9. (10 pts.) The following statements relate to King, et al., "Is Proof More Cost Effective than Testing?" Indicate whether each is true or false. To compensate for random guessing, your score in points will be 2 times the number of [correct minus incorrect] answers, or 0 – whichever is greater. Therefore, if you are not more than 50% sure of your answer, consider skipping the problem.

a. The information system described in the paper, SHOLIS, was selected to evaluate the feasibility of the formal techniques mandated by UK Defense Standard 00-55 before they were to be employed in the development of any UK Ministry of Defense safety critical systems.	true	false
b. The executable part of the programming language used is a subset of Ada, but permits additional annotations that make it possible to prove partial code correctness using a commercial toolset.	true	false
c. The development process used for SHOLIS was "a fairly standard one" except that requirements were written using Z instead of English.	true	false
d. In contrast to proofs at the Z level, all the code level proof work was carried out with machine assistance.	true	false
e. All of the proof engineers were experienced mathematicians and software engineers who had worked for several years in various formal methods.	true	false

10. (10 pts.) The following statements relate to Linger, "Cleanroom Software Engineering for Zero-Defect Software." Indicate whether each is true or false. To compensate for random guessing, your score in points will be 2 times the number of [correct minus incorrect] answers, or 0 – whichever is greater. Therefore, if you are not more than 50% sure of your answer, consider skipping the problem.

a. In the Cleanroom process, team correctness verification takes the place of unit testing and debugging; there is no software execution by the development team.	true	false
b. The system-level test team is responsible for "testing-in" quality using statistical usage testing.	true	false
c. Linger notes that while the Cleanroom process is readily applied to new systems development, re-engineering and extension of existing systems require the use of other, more traditional processes.	true	false
d. Quality results from completed Cleanroom software projects are not reported in the paper, but are described by Linger as being "very encouraging".	true	false
e. Cleanroom's statistical usage testing is biased by failure-rate in finding errors.	true	false

11. a. (3 pts.) For which one of the following do Dunlop and Basili ("A comparative Analysis of Functional Correctness") use the shorthand notation "*B∗H*" ? (Circle ONE only.)

i. a while loop program with total predicate *B* on state *X,* and body function *H*

ii. the set of data states associated with the Cartesian product of functions *B* and *H*

iii. the logical conjunction of predicates *B* and *H*

iv. the Bounded Hierarchy of internal invariants for the function of a looping program

v. the Bradley and Hubbard Subgoal Induction verification technique

b. (5 pts.) Recall that in Problem Set 7 we considered the question, *"Are there (intended program) functions that cannot be computed by while loops?"* Consider the function ***f* = (x,y := x,y+x)**. Prove that there is NO program of the form *while b do S* that computes *f* for (all) X∈D(*f*) based on the proof method used in the Problem Set 7 Solution Notes.

On my honor, I have neither given nor received unauthorized aid on this exam and I pledge not to divulge information regarding its contents to those who have not yet taken it.

_____

SIGNATURE